Few-Shot Testing of Autonomous Vehicles With Scenario Similarity Learning

Shu Li, Honglin He, Jingxuan Yang, Jianming Hu, *Senior Member, IEEE*, Yi Zhang, *Senior Member, IEEE*, and Shuo Feng, *Member, IEEE*

***Abstract*—Testing and evaluation are critical to the development and deployment of autonomous vehicles (AVs). Given the rarity of safety-critical events such as crashes, millions of tests are typically needed to accurately assess AV safety performance. Although techniques like importance sampling can accelerate this process, it usually still requires too many tests for field testing. This severely hinders the testing and evaluation process, especially for third-party testers and governmental bodies with very limited testing budgets. The rapid development cycles of AV technology further exacerbate this challenge. To fill this research gap, this paper introduces the few-shot testing (FST) problem and proposes a methodological framework to tackle it. As the testing budget is very limited, usually smaller than 100, the FST method transforms the testing scenario generation problem from probabilistic sampling to deterministic optimization, reducing the uncertainty of testing results. To optimize the selection of testing scenarios, a cross-attention similarity mechanism is proposed to extract the information of AV's testing scenario space. This allows iterative searches for scenarios with the smallest evaluation error, ensuring precise testing within budget constraints. Experimental results in cut-in scenarios demonstrate the effectiveness of the FST method, significantly enhancing accuracy and enabling efficient, precise AV testing.**

***Index Terms*—Few-shot testing, autonomous vehicles, scenario similarity, deep learning.**

I. Introduction

TESTING and evaluation of autonomous vehicles(AVs) has attracted great interest from researchers in recent years [1], [2], [3], [4], [5]. The underlying significance of testing and evaluation for AVs arises from the safety-critical nature of open-road applications. However, the rarity of safety-critical events (e.g. crashes) within seemingly endless traffic scenarios in real world [6] substantially undermines the efficacy of testing the performance index of AVs [7]. Consequently, there is a compelling imperative to judiciously generate testing scenarios to enhance the efficiency of AV evaluation processes.

During the extensive discussions of AV testing with advanced intelligent methods, approaches such as accelerated testing [8] and corner-case evaluation [9] have been primarily used to assess AV performance under sufficient testing budgets. However, no widely accepted approach has been established for routine evaluation across different AVs. We suggest that this is primarily due to the extremely limited testing budgets in many real-world applications, which render existing methods ineffective. For example, third-party testing organizations and governmental bodies cannot test a large set of scenarios for all potential AV models, especially when real AV testing is required. Besides, with the rapid iterative development of autonomous driving techniques, conducting a thorough evaluation of AV performance within the research and development cycle also becomes increasingly infeasible. In these realistic cases, a preliminary yet reliable testing and evaluation result is urgently needed, and the result must be generated within the confines of an extremely small budget for testing. The testing procedure should also be quite concise and deterministic, thus being able to expediently generalize among numerous possible AVs under test. Moreover, quantitative and explainable testing results are needed to compare the performance of different AVs, which creates additional difficulties. In practice, corner cases are often generated by experts [10] or replayed from logged data [11]. However, these heuristic approaches lack a solid theoretical foundation, and unknown unsafe scenarios can still be missed.

In this paper, we formulate this problem as the few-shot testing (FST) problem and propose the FST method to tackle this problem. To the best of our knowledge, this is the first time the FST problem is developed and attacked.

Current testing methods have failed to address the FST problem because they cannot accurately quantify the performance index of AVs or control the substantial uncertainty and variance within an acceptably small range when the testing budget is extremely limited (e.g., fewer than 10^2 scenarios). In light of these failures, we applied statistical models to quantify the AV performance index. Furthermore, to eliminate the uncertainty caused by the statistical sampling method, we transform the testing problem from a probabilistic sampling problem to a deterministic optimization problem, as shown in Fig. 1.

Received 29 August 2024; revised 7 March 2025 and 15 August 2025; accepted 22 September 2025. This work was supported in part by the National Natural Science Foundation of China under Grant 62473224, in part by Beijing Nova Program under Grant 20240484642 and Grant 20230484259, and in part by Beijing Natural Science Foundation under Grant 4244092. The Associate Editor for this article was C. Ahlstrom. *(Corresponding author: Shuo Feng.)*

Shu Li, Honglin He, Jingxuan Yang, and Jianming Hu are with the Department of Automation, Tsinghua University, Beijing 100084, China (e-mail: li-s23@mails.tsinghua.edu.cn; hehl21@mails.tsinghua.edu.cn; yangjx20@mails.tsinghua.edu.cn; hujm@tsinghua.edu.cn).

Yi Zhang is with the Department of Automation, Beijing National Research Center for Information Science and Technology (BNRist), Tsinghua University, Beijing 100084, China, and also with Tsinghua-Berkeley Shenzhen Institute (TBSI), Shenzhen 518055, China (e-mail: zhyi@mail.tsinghua.edu.cn).

Shuo Feng is with the Department of Automation, Beijing National Research Center for Information Science and Technology (BNRist), Tsinghua University, Beijing 100084, China (e-mail: fshuo@tsinghua.edu.cn).

Digital Object Identifier 10.1109/TITS.2025.3622257

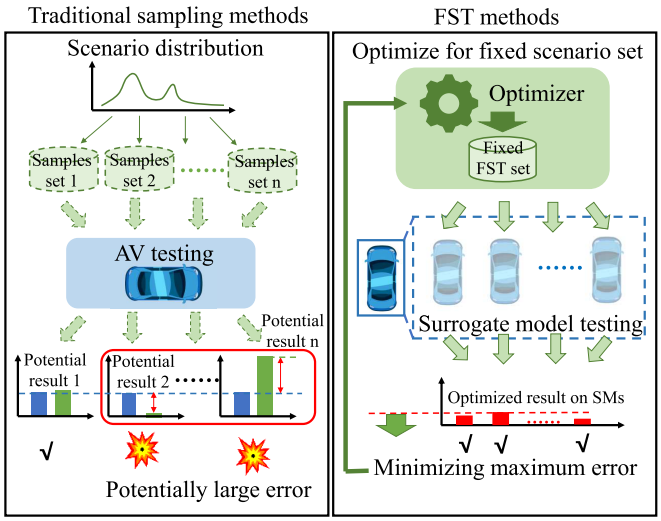

Fig. 1. The comparison of the basic idea between statistical sampling methods and the FST method.

We prove the optimality and feasibility of this deterministic optimization and use it to identify an FST scenario set with strong generalization ability. This improves the applicability of the FST method for testing different AVs in real-world scenarios. To leverage the available testing budget, the FST scenario set is searched from a global perspective rather than sampled sequentially. Finally, using the fixed scenario set, we assess the precision of FST method based on the fixed evaluation error. With some information on AVs, a minimum upper bound of the evaluation error can be derived to ensure the reliability of FST scenarios.

Specifically, we employ surrogate models (SMs, as in [12], [13]) to represent possible AVs under test. To extract information from the scenario space of AVs, we propose a cross-attention similarity network to learn the relationships between selected FST scenarios and other scenarios. The similarity network quantifies the global information gain of testing scenarios and facilitates the fusion of AV testing results to obtain the final evaluation result. After training the similarity network, we use gradient descent optimizer to conduct a global, iterative search for a FST scenario set with optimal generalization ability and a minimized upper bound of evaluation errors. Ultimately, our method enables the generation of an optimized scenario utilization strategy with optimized FST scenarios.

We designed experiments on the cut-in case (commonly used in previous studies [13], [14], [15] on AV testing) to test and evaluate the performance index of AVs with an extremely small number of scenarios ($n = 5, 10, 20$). To the best of our knowledge, such a small number of testing scenarios has not been employed in previous studies. Experimental results show that the proposed FST method significantly outperforms existing approaches in terms of accuracy. Even under a strictly limited testing budget, the relative error of the FST method remains within an acceptably small range, offering a novel opportunity for rapid and reliable AV testing.

The contribution of this paper can be summarized as follows:

(1) We formulate the FST problem and comprehensively analyze its features and underlying challenges compared to traditional testing and evaluation problems.

(2) We propose a general FST framework and theoretically analyze its optimality, feasibility and sensitivity. The performance index of the AV can be evaluated with an upper bound of estimation errors.

(3) We devise a deep learning framework as an implementation of the FST method. It is capable of learning to extract the scenario similarity while keeping the theoretical benefits.

(4) We conduct systematic experiments to verify the effectiveness of the FST method in terms of accuracy, stability, sensitivity, and convergence, and the results demonstrate a significant improvement compared to prevailing AV testing methods.

II. Related Works

Many efforts have been made to search for a smaller testing scenario set or accelerate testing process from different perspectives. Although these methods cannot be applied directly to FST cases, we would still provide a review of these methods and analyze their limitations.

Using critical or risky scenarios to test AVs is intuitively easier to discover defects and reduce testing costs. As a practical method, some autonomous driving companies maintain a scenario set from logged data and expert knowledge to verify the reliability of their AVs before on-road deployment [11]. Searching for critical scenarios or corner cases is also a commonly used scheme to generate a smaller testing scenario set [16]. Based on knowledge [10], scenario clustering [17], [18], scenario coverage [19], optimization strategy [20], [21] or other carefully designed models [15], [22], [23], [24], many methods are capable of generating a representative scenario set with certain risks. Deep learning [25] and reinforcement learning [26] methods are also effective tools for scenario generation. Given a small testing budget, these methods will be able to generate specific testing scenarios. However, the efficiency of these scenarios is usually measured using risk, realism, or other specially designed metrics. In addition to being unable to quantify the performance index of AVs, the frequent occurrence of unknown risky scenarios also challenges the effectiveness of these methods in providing reliable AV evaluations.

Statistical sampling methods represent an effective approach to quantifying the performance index of AV model while generating critical scenarios to accelerate the testing process [8], [13], [14], [27], [28], [29]. Based on naturalistic driving data (NDD), the naturalistic driving environment (NDE) can be constructed and is widely used in these statistical testing methods. Furthermore, the performance of AVs can be estimated with a critical distribution using importance sampling (IS). These methods can generate unbiased quantitative results with higher efficiency. However, the critical scenarios may be similar or repetitive, potentially resulting in redundant AV information. As illustrated in Fig. 1, the testing variance in an extremely small test set becomes extremely large. Since only a single set of samples can be tested, it is almost impossible

TABLE I
SUMMARY OF NOTATION

Notation	Definition	Notation	Definition
n	number of the FST scenarios	m	surrogate model
$\mathcal{X}$	scenario space	P_m	performance of m
$\mathcal{X}_n$	set of n FST scenarios	μ_m	ground truth of performance index of model m
$\mathcal{X}_i$	subset of scenario space $\mathcal{X}$	m^*	AV model
$\boldsymbol{x}_i$	i-th FST scenario	w	weight function
$\boldsymbol{x}, \boldsymbol{x}_{(j)}$	(j-th) scenario in $\mathcal{X}$	S	similarity function
$\boldsymbol{x}_{i,\mathrm{CMC}}$	i-th CMC scenario	$\boldsymbol{Q}, \boldsymbol{K}, \boldsymbol{V}$	query, key, value matrix
$\boldsymbol{X}_*$	random variable for $\boldsymbol{x}_*$	$\boldsymbol{W}$	weight matrix
p	probability of scenario	$\boldsymbol{S}$	similarity matrix
A	event of interest (crash)	θ	network parameters
P	performance index function	L	loss function
$\mathbb{E}$	mathematical expectation operator	P_c	critical distribution of $\mathcal{X}_n$
$\mu_{(*)}$	performance ground truth (of model $*$, AV by default)	Δm	error model between AV and $\mathcal{M}$
$\tilde{\mu}_{\mathrm{CMC}}$	μ estimated by CMC	$\hat{\mu}_{F,\mathrm{FST}}$	estimation of $\tilde{\mu}_{\Delta m,\mathrm{FST}}$ with fluctuation
$\tilde{\mu}_{(*),\mathrm{FST}}$	μ estimated by FST (on model $*$, AV by default)	$E_{\max}$	maximum value of E on $\mathcal{M}$
E	error of FST	F	fluctuation estimator
f	testing results fusion function of FST	w_F	weight parameter of fluctuation estimator
$\mathcal{M}$	surrogate model set		

to determine whether the obtained results happen to provide an accurate estimation of the AV performance.

The above methods generate critical or risky scenarios based on offline strategies. By leveraging bayesian optimization or regression methods, it is also possible to adaptively testing the performance index of AV, thus utilize potential information of AVs [12], [30], [31], [32]. However, in FST cases where the testing budget is extremely small, the extra information from the AV is insufficient to improve testing and evaluation efficiency and accuracy. Meanwhile, since these methods generate scenarios incrementally, some global information about the AV may be overlooked, making it difficult to ensure accuracy with such a small number of scenarios.

In [33], we proposed a coverage-based FST method (FST-C). With the handcrafted coverage model, testing precision could be significantly improved compared with previous methods. However, its effectiveness is highly dependent on the design of the coverage model and the AV, which may limit its applicability to different AVs or scenarios. In many specific cases, the error of the handcrafted method is not small enough to yield useful and accurate testing and evaluation results. The goal of developing a general few-shot testing method applicable to diverse AVs and scenarios has yet to be achieved.

III. Problem Formulation

In this section, we provide the formulation of the general FST problem. The notation used in our paper are listed in Table I.

A. Performance Index Testing

To quantify the performance index (e.g. crash rate) of AV under test, we use the NDE to model the driving environment of AVs, which is a general formulation and is applied in many existing studies [8], [27], [28]. In the NDE, the testing state space $\mathcal{X}$ is restricted by the operational design domain. The exposure frequency and the testing performance of AV on certain scenario $\boldsymbol{x} \in \mathcal{X}$ is defined as $p(\boldsymbol{x})$ and $P(A|\boldsymbol{x})$. We consider $P(A|\boldsymbol{x})$ as the probability of the event of interest A (e.g. crashes) on scenario $\boldsymbol{x}$. Then $P(A)$, which means the overall performance index of AV and is also the overall probability of event A, is defined as follows:

$$P(A) = \sum_{\boldsymbol{x} \in \mathcal{X}} P(A|\boldsymbol{x}) p(\boldsymbol{x}). \tag{1}$$

From the perspective of the random variable $\boldsymbol{X}$ taking concrete values $\boldsymbol{x} \in \mathcal{X}$ with the probability distribution $p(\boldsymbol{x})$, Eq. (1) can also be written as

$$\mu = \mathbb{E}_p[P(A|\boldsymbol{X})], \tag{2}$$

where $\mu = P(A)$ is the mathematical expectation of $P(A|\boldsymbol{X})$ and serves as the ground truth of the performance index of AV under test.

For traditional methods that test AV directly in the NDE, testing scenarios are generated with Crude Monte Carlo (CMC) [34]. The CMC method samples a testing scenario set $\mathcal{X}_{n,\mathrm{CMC}} \triangleq \{\boldsymbol{x}_{1,\mathrm{CMC}}, \ldots, \boldsymbol{x}_{n,\mathrm{CMC}}\}$ from the original distribution $p(\boldsymbol{x})$, and the expectation of $P(A|\boldsymbol{X})$ is estimated by

$$\tilde{\mu}_{\mathrm{CMC}} = \frac{1}{n} \sum_{i=1}^{n} P(A|\boldsymbol{x}_{i,\mathrm{CMC}}), \boldsymbol{X}_{i,\mathrm{CMC}} \sim p(\boldsymbol{X}). \tag{3}$$

As $n \to \infty$, it can be proved that $\tilde{\mu}_{\mathrm{CMC}}$ converges to μ with the probability of 1. Therefore CMC is unbiased and can provide a reliable estimation of AV performance index with a sufficiently large number of tests. According to reports [35], millions or even billions of miles of test are required to demonstrate the reliability of AVs in fatal crashes and this is almost impossible in real world application. As a contrast, when the number of scenarios n is limited within a very small range (such as $n \leq 10^2$), the estimation variance of CMC becomes almost impossible to control.

B. Target of Few-Shot Testing

The fundamental testing objective of the FST problem is the testing accuracy on AVs under a strictly limited number of scenarios n. In this study, we tackle the testing problem with the idea of optimizing instead of sampling from distributions. Due to the substantial uncertainty when n is small, sampling methods typically result in a large variance. In contrast, we search

for a fixed testing scenario set $\mathcal{X}_{n,\mathrm{FST}} = \{\boldsymbol{x}_{1,\mathrm{FST}}, \ldots, \boldsymbol{x}_{n,\mathrm{FST}}\}$ given a specific n, and we use $\mathcal{X}_n$ and $\boldsymbol{x}_i$ for short in the remainder of this paper. We try to minimize the evaluation error compared to the ground truth of AV performance:

$$\min_{\mathcal{X}_n} E = |\tilde{\mu}_{\mathrm{FST}} - \mu| . \tag{4}$$

In Eq. (4), $\tilde{\mu}_{\mathrm{FST}}$ is a fixed value after the testing set $\mathcal{X}_n$ is determined. The evaluation error will then be a certain value with no variance. With this scheme, we transform the problem of minimizing variance using unbiased sampling method into minimizing fixed errors with a set of fixed testing scenarios. Compared to statistical sampling methods, the advantages of this transformation when n is extremely small mainly lie in:

(1) with a fixed and optimized FST set, the uncertainty is eliminated, which ensures the accuracy and reliability of the FST method in cases where the testing budget is strictly limited;

(2) all scenarios are selected collectively using a high-level strategy, rather than being generated sequentially or independently from distributions, thereby maximizing the utility of each testing scenario from a global perspective.

C. General Few-Shot Testing Problem

Generally, the estimation result of the FST method is a function of the testing scenarios and we can rewrite Eq. (4) with

$$\tilde{\mu}_{\mathrm{FST}} = f\left[P(A|\boldsymbol{x}_1), \ldots, P(A|\boldsymbol{x}_n)\right] \tag{5}$$

to get the expanded form of FST target

$$\min_{\mathcal{X}_n} E = |f\left[P(A|\boldsymbol{x}_1), \ldots, P(A|\boldsymbol{x}_n)\right] - \mu| . \tag{6}$$

In Eq. (6), however, it is impossible to get all information on AV (namely $P(A|\boldsymbol{x}_i)$ and μ) before testing to solve this optimization problem. In practical terms, only part of the prior knowledge about the AV is known. Moreover, since the FST set is fixed after optimization and the AV under test is unknown, FST method must have strong generalization ability across potential AVs. Consequently, we suppose that the possible AVs forms a vehicle model set $\mathcal{M}$. For all possible models $m \in \mathcal{M}$, the performance index of AV in scenario $\boldsymbol{x}$ can be tested as $P_m(A|\boldsymbol{x})$. The SM set $\mathcal{M}$ not only provide information of the AVs, but also captures the uncertainty of them. Moreover, the SM set could be continuously updated to better cover the diversity of the real-world AV behaviors. As FST method is devised to generate accurate results with minimized errors, we can further formulate Eq. (6) as

$$\min_{\mathcal{X}_n} \max_{m\in\mathcal{M}} E = |f\left[P_m(A|\boldsymbol{x}_1), \ldots, P_m(A|\boldsymbol{x}_n)\right] - \mu_m| , \tag{7}$$

where $\mu_m = \mathbb{E}_p[P_m(A|X)]$ is the ground truth of performance index of specific model m and E is the testing and evaluation error.

We can see from Eq. (7) that n scenarios are carefully selected to extract the performance of all possible AVs under test. Supposing the real AV under test m^* satisfies $m^* \in \mathcal{M}$, the accuracy and reliability of FST results are ensured by an upper bound of error, thus addressing the substantial uncertainty and unacceptable confidence levels arise from an extremely small n. The accuracy of prior information may vary across different testing problems, while this formulation can always optimize the performance by leveraging the available data to its fullest extent.

Apart from these advantages, the challenges to solving this problem can be summarized as follows:

(1) the form of the estimation function f is highly flexible and the method to generate $\tilde{\mu}_{\mathrm{FST}}$ using testing performances $P(A|\boldsymbol{x}_i), i = 1, \ldots, n$ remains undetermined;

(2) supposing a concrete f is decided, a set of scenarios $\mathcal{X}_n$ should be carefully selected to obtain a minimized upper bound of the evaluation error;

(3) if $\mathcal{M}$ contains additional errors and $m^* \in \mathcal{M}$, the effectiveness of the testing and evaluation results may be impaired.

Our solutions to these challenges will be discussed in the next section.

IV. Few-Shot Testing Method

A. Vehicle Model Set Construction

As a representation of the prior knowledge of possible AVs, we use surrogate models (SMs) to construct $\mathcal{M}$ in Eq. (7) in this paper. In previous works [12], [13], SM has commonly been used as an effective way to draw a sketch on AV models for further testing and evaluation. For the FST problem, we focus on the generalization ability of the FST method on various potential AVs to get minimized errors, so we use multiple SMs $m_1, \ldots, m_s$ to form a surrogate model set, which is $\mathcal{M}$. For simplicity, we assume that

$$\mathcal{M} \triangleq \left\{ m | m = \sum_{i=1}^{s} c_i m_i, c_i > 0, \sum_{i=1}^{s} c_i = 1 \right\}, \tag{8}$$

which means the AV model m^* can be approximated as a linear combination of s possible SMs (the combination is based on the performance level, i.e. $P_m(A|\boldsymbol{x})$). These SMs could consist of vehicle models from aggressive driving policies to conservative driving policies and can depict the possible strategies of AVs. Eq. (8) is not the only possible description of M. For instance, SMs with different noise magnitudes in the scenario space can also form a set M. The noise magnitudes indicate the confidence we have in the prior knowledge of AVs, which could be explored in future work.

With this simple form of the SM set, the optimization problem in Eq. (7) will be easier to solve and we have the following theorem:

Theorem 1: The following 2 descriptions of minimax problem are equivalent under Eq. (8):

$$\min_{\mathcal{X}_n} \max_{m\in\mathcal{M}} |f\left[P_m(A|\boldsymbol{x}_1), \ldots, P_m(A|\boldsymbol{x}_n)\right] - \mu_m| , \tag{9a}$$

$$\min_{\mathcal{X}_n} \max_{i=1,\ldots,s} \left| f\left[P_{m_i}(A|\boldsymbol{x}_1), \ldots, P_{m_i}(A|\boldsymbol{x}_n)\right] - \mu_{m_i} \right| . \tag{9b}$$

Proof: For fixed $\mathcal{X}_n$, we focus on the maximization problem. Assume that the optimal solution to the maximization problem in Eq. (9a) is $\boldsymbol{c}^* = (c_1^*, \ldots, c_s^*)$ with the optimal value E^* and there exists $j \neq k$ such that $0 < c_j^*, c_k^* < 1$. For another two solutions $\boldsymbol{c}'$ and $\boldsymbol{c}''$, taking

$$c_i' = c_i'' = c_i^*, \forall i \neq j, k,$$

$$c'_j = c''_k = c^*_j + c^*_k,$$
$$c'_k = c''_j = 0,$$

and denoting E', E'' as the objective function value respectively, we have

$$E^* = \frac{c^*_j}{c^*_j + c^*_k} E' + \frac{c^*_k}{c^*_j + c^*_k} E'' \leq \max\{E', E''\}.$$

Thus $\boldsymbol{c}^*$ is not the optimal solution and c^*_j or c^*_k can be converted to 0 to reach $\boldsymbol{c}'$ or $\boldsymbol{c}''$. Repeat this operation and the optimal value can be taken only at $\boldsymbol{c}^*$ where $c^*_i = 1, c^*_j = 0, \forall j \neq i$, which is the form in Eq. (9b).

Consequently, with this construction of the SM set, the complex optimization problem in Eq. (7) is simplified. This will be beneficial when we solve this optimization problem in experiments.

B. Testing Results Fusion

In this section, we try to solve the problem of fusing the testing results on AV, namely $P(A|\boldsymbol{x}_i), \ldots, P(A|\boldsymbol{x}_n)$, to evaluate $\tilde{\mu}_{\text{FST}}$ (see Eq. (5)). Classic testing method CMC provides us with a direct way in Eq. (3), where all scenarios sampled from the distribution $p(\boldsymbol{x})$ are assigned the same weight $1/n$. In this paper, we extend this form to a more general weighted sum to evaluate the contribution of each FST scenario (the testing results) to the overall evaluation result

$$\begin{aligned}\tilde{\mu}_{\text{FST}} &= f[P(A|\boldsymbol{x}_i), \ldots, P(A|\boldsymbol{x}_n)] \\ &= \sum_{i=1}^{n} P(A|\boldsymbol{x}_i) w(\boldsymbol{x}_i; \mathcal{X}_n),\end{aligned} \tag{10}$$

where $w(\boldsymbol{x}_i; \mathcal{X}_n)$ is the weight assigned to each testing result and is related to all testing scenarios in $\mathcal{X}_n$. To obtain a normalized evaluation result $\tilde{\mu}_{\text{FST}}$, we assume that

$$\sum_{i=1}^{n} w(\boldsymbol{x}_i; \mathcal{X}_n) = 1. \tag{11}$$

Comparing to statistical methods like CMC and IS [8], [14], [29], which suffer from high uncertainty and large variance when the number of scenarios n is small, our testing results fusion strategy adapts better to FST problems. As w is a flexible value in the range $[0, 1]$, the weight of each scenario to the evaluation result is adjusted based on the information it provides, thus ameliorating the problem of information redundancy or insufficiency in statistical testing methods. Additionally, w is a function of the entire testing set $\mathcal{X}_n$. The weight of each scenario is determined jointly rather than independently, which maximizes the utility of the small number of testing scenarios.

In the formulation of the FST problem, we aimed to find the set of optimized scenarios with the fixed and smallest error. Consequently, the optimality of the testing result fusion strategy is important for the FST method. For the FST method defined with Eq. (10-11), we can prove in the following theorem that for any AV model m, there exists a weight function w and an FST set $\mathcal{X}_n$ such that the evaluation error on m is 0.

Theorem 2: Let μ be the performance index of model m and $\mathcal{X}_n$ be the FST scenario set of n. $\tilde{\mu}_{\text{FST}}$ is given by Eq. (10). Then there exists a weight function $w(\boldsymbol{x}_i; \mathcal{X}_n)$ satisfying Eq. (11) so that

$$\tilde{\mu}_{\text{FST}} = \mu. \tag{12}$$

Proof: Given fixed $\boldsymbol{x}_i$ and $P(A|\boldsymbol{x}_i)$, we can write $\tilde{\mu}_{\text{FST}}$ as

$$\begin{aligned}\tilde{\mu}_{\text{FST}} &= \sum_{i=1}^{n} P(A|\boldsymbol{x}_i) w(\boldsymbol{x}_i; \mathcal{X}_n), \\ &= g(w_1, \ldots, w_n),\end{aligned}$$

where $w_i = w(\boldsymbol{x}_i; \mathcal{X}_n)$ for short. According to the definition of μ we have $\min_{i=1,\ldots,n} P(A|\boldsymbol{x}_i) \leq \mu \leq \max_{i=1,\ldots,n} P(A|\boldsymbol{x}_i)$. Then define $j = \text{argmin}_j P(A|\boldsymbol{x}_j)$ and $k = \text{argmax}_k P(A|\boldsymbol{x}_k)$. Given $w_j = 1$ and $w_i = 0, \forall i \neq j$, we have

$$\tilde{\mu}_{\text{FST}} = g(w_1, \ldots, w_n) = P(A|\boldsymbol{x}_j) \leq \mu$$

and given $w_k = 1$ and $w_i = 0, \forall i \neq k$, we have

$$\tilde{\mu}_{\text{FST}} = g(w_1, \ldots, w_n) = P(A|\boldsymbol{x}_k) \geq \mu.$$

With the arbitrariness of $w_1, \ldots, w_n$ we have the continuity of $g(w_1, \ldots, w_n)$ and Eq. (12) can be acquired.

With Theorem 2, we prove the optimality and feasibility of the FST method. Then it is significant to find a weight function to achieve Eq. (12). Following the insight of extracting representative information from the FST scenario set, we propose the similarity measure as an effective representation of the information contained in scenarios, denoted as

$$w(\boldsymbol{x}_i; \mathcal{X}_n) = \sum_{\boldsymbol{x} \in \mathcal{X}} S(\boldsymbol{x}_i, \boldsymbol{x}; \mathcal{X}_n) p(\boldsymbol{x}), \tag{13}$$

where $S(\boldsymbol{x}_i, \boldsymbol{x}; \mathcal{X}_n)$ is the similarity between a FST scenario $\boldsymbol{x}_i$ and another scenario $\boldsymbol{x}$ in the entire state space. To fully leverage the global information from an extremely small set of scenarios, the similarity $S(\cdot)$ between any two scenarios $\boldsymbol{x}_i$ and $\boldsymbol{x}$ is influenced by the entire FST set $\mathcal{X}_n$, which enables a potentially higher evaluation accuracy.

In Eq. (13) the weight of the FST scenario $\boldsymbol{x}_i \in \mathcal{X}_n$ is interpreted as the sum of its similarity to scenarios in the state space, weighted by the exposure frequency of the other scenario. If the similarity measure satisfies

$$\sum_{i=1}^{n} S(\boldsymbol{x}_i, \boldsymbol{x}; \mathcal{X}_n) = 1, \forall \boldsymbol{x} \in \mathcal{X}, \tag{14}$$

combining with the fact $\sum_{\boldsymbol{x} \in \mathcal{X}} p(\boldsymbol{x}) = 1$ and substituting Eq. (14) into Eq. (13), we can know that it satisfies our normalization condition in Eq. (11).

Intuitively, it is reasonable to decide the contribution of each FST scenario according to the number of other scenarios that are similar to it. The concept of similarity is also utilized in various methods of AV testing [36] or scenario clustering [6]. Typically, similarity is represented by a pre-defined, handcrafted model (e.g. distance in the state space) and remains fixed in the testing process. As similarity is a reflection or hypothesis on the features of the state space, its effectiveness is closely tied to the specific AV under test. As a result, the reliability of the handcrafted model is in question.

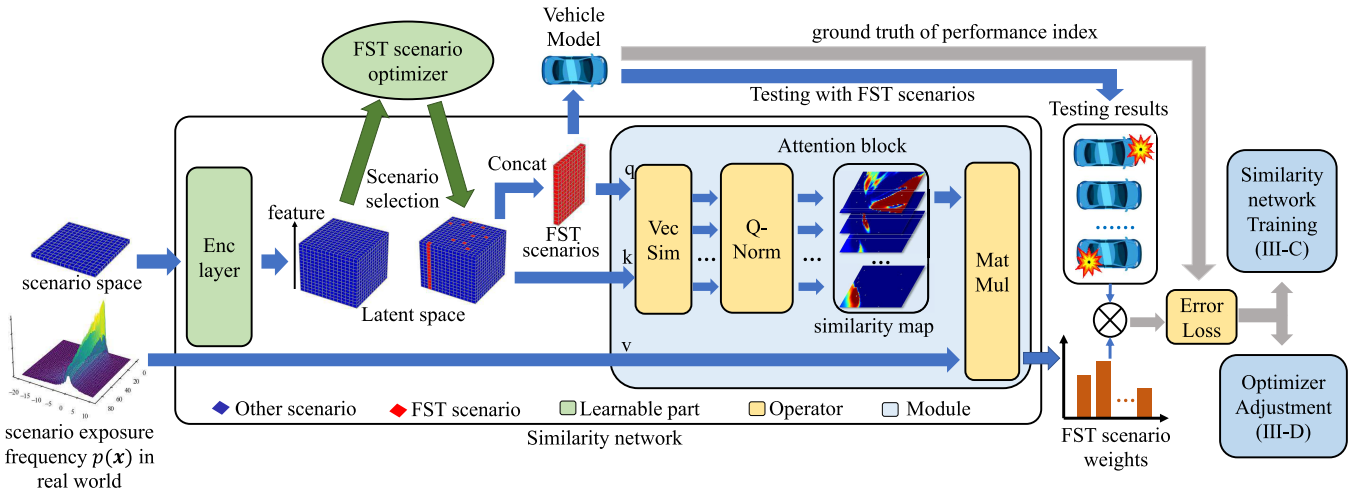

Fig. 2. The structure of cross-attention similarity network and FST framework.

C. Scenario Similarity Learning

Based on the testing result fusion strategies from Eq. (10) to (14), we propose a general learnable similarity measure. Our purpose is to design a similarity function $S(\boldsymbol{x}_i, \boldsymbol{x}; \mathcal{X}_n)$ satisfying Eq. (14) so that the upper bound of the testing and evaluation error E in Eq. (7) is minimized. To leverage the potential data provided by SMs, we propose a cross-attention similarity network in a self-supervised learning manner. The architecture of the framework is illustrated in Fig. 2.

Instead of directly applying the similarity model to different scenarios, we use the MLP to extract latent features of all possible scenarios based on the surrogate model set $\mathcal{M}$. The core of the similarity network is to generate a similarity measure between a scenario $\boldsymbol{x_i} \in \mathcal{X}_n$ in the FST scenario set and a scenario $\boldsymbol{x} \in \mathcal{X}$ in the scenario space. This similarity value is supposed to be adjusted based on $\mathcal{X}_n$ to utilize global information effectively. Therefore, treating each scenario as a token, we propose a cross-attention structure between the FST scenarios $\boldsymbol{x}_i$ and general scenarios $\boldsymbol{x}$. All encoded samples in the FST scenarios set $\boldsymbol{x}_i$ are used as queries to calculate attention to other encoded scenarios $\boldsymbol{x}$, which serve as keys. In order to dynamically adjust the attention mechanism according to the FST scenarios in $\mathcal{X}_n$, we apply the query-level normalization, which normalizes the weights of values assigned to queries along the dimension of queries. This normalization strategy extracts the relative information of the FST scenario set $\mathcal{X}_n$. The similarity matrix is generated as

$$\boldsymbol{S}_{n*N} = \mathrm{Norm}_q \left\{ d(\boldsymbol{Q}, \boldsymbol{K}) \right\}, \tag{15}$$

where $\boldsymbol{Q}_{n\times d}$ and $\boldsymbol{K}_{N\times d}$ is the encoded feature matrix from $\mathcal{X}_n$ and $\mathcal{X}$. N is the number of scenarios in state space. $d(\boldsymbol{Q}, \boldsymbol{K})$ is a feature-wise similarity calculation strategy (e.g. $\boldsymbol{Q}\boldsymbol{K}^T$ for classic attention). Here we compute the reciprocal of the L2 distance between each column in $\boldsymbol{Q}$ and $\boldsymbol{K}$ to get $d(\boldsymbol{Q}, \boldsymbol{K})_{n\times N}$. Softmax is used as the normalization function along the query dimension. The element s_{ij} in $\boldsymbol{S}_{n*N}$ means the similarity between the i-th FST sample and the j-th scenario in scenario space, i.e. $S(\boldsymbol{x}_i, \boldsymbol{x}_{(j)}; \mathcal{X}_n)$. To improve efficiency, $N' < N$ scenarios in state space $\mathcal{X}$ can also be used as References to compute similarity $\boldsymbol{S}_{n*N'}$. Finally, according to Eq. (13), we merge $p(\boldsymbol{x})$ to get matrix $\boldsymbol{V}_{N\times 1}$ and use the attention output of FST scenario tokens as the weight function

$$\boldsymbol{W}_{n\times 1} = \boldsymbol{S}_{n\times N} \boldsymbol{V}_{N\times 1}, \tag{16}$$

where $\boldsymbol{W}$ can be expanded as $[w(\boldsymbol{x}_i; \mathcal{X}_n), \ldots, w(\boldsymbol{x}_n; \mathcal{X}_n)]^T$. With the cross-attention mechanism, the weight w and estimation result $\tilde{\mu}_{FST}$ can be calculated to train the similarity network. Note that with continuous similarity network parameters $\boldsymbol{\theta}$ and the query-level normalization, Eq. (14) is satisfied, so the optimal testing and evaluation error is 0 for all $m \in \mathcal{M}$ with optimal θ, according to Theorem 2.

After obtaining weight matrix in Eq. (16) we can calculate the FST result according to Eq. (10) and subsequently calculate the upper bound of the evaluation error in Eq. (7). We directly use the upper bound of the error as the loss function. Since the testing result fusion function f is derived from the similarity network, we denote it as f_θ and have

$$L(\boldsymbol{\theta}, \mathcal{X}_n) = \max_{m\in\mathcal{M}} \left| f_{\boldsymbol{\theta}} \left[P_m(A|\boldsymbol{x}_1), \ldots, P_m(A|\boldsymbol{x}_n) \right] - \mu_m \right|. \tag{17}$$

Generally, the similarity network is trained by randomly sampled $\mathcal{X}_n$ to get minimized expectation of loss $\mathbb{E}_{\mathcal{X}_n}[L(\boldsymbol{\theta}, \mathcal{X}_n)]$. However, if we consider the optimization of $\mathcal{X}_n$ in Eq. (7), the optimization target will be

$$\min_{\boldsymbol{\theta}, \mathcal{X}_n} L(\boldsymbol{\theta}, \mathcal{X}_n), \tag{18}$$

which means we should optimize $\boldsymbol{\theta}$ and $\mathcal{X}_n$ simultaneously. Eq. (18) can be interpreted as prioritizing the loss value at the optimal point $\mathcal{X}_n^*$ rather than closely monitoring the average loss value on other FST scenarios.

Because $\mathcal{X}_n^*$ is completely unknown before training, and the optimization of $\mathcal{X}_n$ is coupled with the optimization of $\boldsymbol{\theta}$, Eq. (18) becomes intractable. We simplify and decouple this problem into a two-stage process: training the similarity network $\boldsymbol{\theta}$ and optimizing the FST set $\mathcal{X}_n$. In the training process, since $\mathcal{X}_n^*$ is unknown, we use a compromised and

practical form to approximate it. We sample $\mathcal{X}_n$ from some critical distribution, written as

$$\min_{\boldsymbol{\theta}} \mathbb{E}_{\mathcal{X}_n \sim P_c}[L(\boldsymbol{\theta}, \mathcal{X}_n)], \tag{19}$$

where P_c is a critical distribution and is supposed to be close to the optimal solution $\mathcal{X}_n^*$. To determine P_c, we apply k-means clustering to the state space based on the performance of SMs. We obtain k subsets of the entire scenario space $\mathcal{X}$, denoted as $\mathcal{X}_i$, where $\mathcal{X}_i \subseteq \mathcal{X}, \forall i$, $\cup_{i=1}^n \mathcal{X}_i = \mathcal{X}$, and $\mathcal{X}_i \cap \mathcal{X}_j = \emptyset, \forall i \neq j$. Each scenario in the training FST scenario set $\mathcal{X}_n$ is uniformly sampled from the subsets $\mathcal{X}_i$ in a circular manner, iterating from 1 to k. This distribution extracts the information of SMs in different areas of the state space and is possible to be an effective approximation to optimal $\mathcal{X}_n^*$.

After the training stage, $L(\boldsymbol{\theta}, \mathcal{X}_n)$ can be used again as the upper bound of the testing and evaluation error in the subsequent generation of the optimal FST scenario set $\mathcal{X}_n$.

D. Testing Scenario Set Optimization

Our remaining step is to optimize for $\mathcal{X}_n$ to achieve the minimum upper bound of error. We write this optimization problem explicitly as

$$\begin{aligned} &\min_{\mathcal{X}_n} \ L_{\boldsymbol{\theta}}(\mathcal{X}_n) \\ &\text{s.t. } L_{\boldsymbol{\theta}}(\mathcal{X}_n) = \max_{m \in \mathcal{M}} |\tilde{\mu}_{\text{FST}} - \mu_m| \\ &\qquad = \max_{m \in \mathcal{M}} \left| \sum_{i=1}^n P_m(A|\boldsymbol{x}_i) w(\boldsymbol{x}_i; \mathcal{X}_n) - \mu_m \right|, \end{aligned} \tag{20}$$

and w is given by Eq. (13). We employ the classic gradient descent method to solve this optimization problem (Fig. 2). The initial $\mathcal{X}_n$ for optimization is sampled with the same strategy P_c as introduced in Section IV-C.

Remarkably, although the similarity network was trained with a specific n, it is capable of adapting to different n. Theoretically, we can write the constraints in Eq. (20) as

$$\begin{aligned} &\left| \sum_{i=1}^n P_m(A|\boldsymbol{x}_i) w(\boldsymbol{x}_i; \mathcal{X}_n) - \mu_m \right| \\ &= \left| \sum_{i=1}^n \left[P_m(A|x_i) \sum_{\boldsymbol{x} \in \mathcal{X}} S_\theta(\boldsymbol{x}_i, \boldsymbol{x}) p(\boldsymbol{x}) \right] - \mu_m \right| \\ &= \left| \sum_{\boldsymbol{x} \in \mathcal{X}} \left\{ \left[\sum_{i=1}^n P_m(A|\boldsymbol{x}_i) S_\theta(\boldsymbol{x_i}, \boldsymbol{x}) \right] - P_m(A|\boldsymbol{x}) \right\} p(\boldsymbol{x}) \right| \\ &\leq \sum_{\boldsymbol{x} \in \mathcal{X}} \left| \left[\sum_{i=1}^n P_m(A|\boldsymbol{x}_i) S_\theta(\boldsymbol{x_i}, \boldsymbol{x}) \right] - P_m(A|\boldsymbol{x}) \right| p(\boldsymbol{x}), \end{aligned} \tag{21}$$

where $\boldsymbol{x}$ is a scenario in the state space. Since $p(\boldsymbol{x}) > 0$ is a constant derived from the dataset, the learning objective is to minimize $\left| \left[\sum_{i=1}^n P_m(A|\boldsymbol{x}_i) S_\theta(\boldsymbol{x_i}, \boldsymbol{x}) \right] - P_m(A|\boldsymbol{x}) \right|$ for all $\boldsymbol{x} \in \mathcal{X}$. If $P_m(A|\boldsymbol{x})$ is close to $P_m(A|\boldsymbol{x}_i)$, this objective would result in a large S_θ for different n.

Once the similarity network is trained, multiple FST sets with different testing numbers n can be obtained by optimization. After n is determined and the FST set is obtained, the FST set can be applied to various AVs to generate accurate testing and evaluation results, making the FST method an efficient and effective testing method.

E. Additional Error Reduction

In this section we present an additional discussion on condition that the real AV under test $m^* \notin \mathcal{M}$, which is possible if $\mathcal{M}$ is still inaccurate. If we decompose m^* into $m^* = m + \Delta m$ where $m \in \mathcal{M}$, we will have

$$\begin{aligned} E^* &= \left| \tilde{\mu}_{m,\text{FST}} + \tilde{\mu}_{\Delta m,\text{FST}} - \mu_m - \mu_{\Delta m} \right| \\ &\leq |\tilde{\mu}_{m,\text{FST}} - \mu_m| + |\tilde{\mu}_{\Delta m,\text{FST}}| + |\mu_{\Delta m}| \\ &\leq E_{\max} + |\tilde{\mu}_{\Delta m,\text{FST}}| + |\mu_{\Delta m}|, \end{aligned} \tag{22}$$

where E^* is testing and evaluation error on m^* and $E_{\max}$ is the upper bound of error given by the FST method on $\mathcal{M}$. In Eq. (22) $|\mu_{\Delta m}|$ is the ground truth performance index of the error model Δm and is a constant value. $|\tilde{\mu}_{\Delta m,\text{FST}}|$ is the testing and evaluation result of Δm with the FST method and can be minimized. We can expand it as

$$\tilde{\mu}_{\Delta m,\text{FST}} = \sum_{i=1}^n \left\{ P_{\Delta m}(A|\boldsymbol{x}_i) \sum_{\boldsymbol{x} \in \mathcal{X}} S(\boldsymbol{x}_i, \boldsymbol{x}; \mathcal{X}_n) p(\boldsymbol{x}) \right\}, \tag{23}$$

where $P_{\Delta m}(A|\boldsymbol{x}_i)$ is the unknown testing result on Δm and can be minimized. As Δm is unknown, we use a fluctuation estimator similar to [33] as an estimation of Δm and $P_{\Delta m}(A|\boldsymbol{x}_i)$:

$$F(\boldsymbol{x}_i; \mathcal{X}_n) \triangleq \frac{\sum_{\boldsymbol{x} \in \mathcal{X}} [P_m(A|\boldsymbol{x}) - P_m(A|\boldsymbol{x}_i)] S(\boldsymbol{x}_i, \boldsymbol{x}; \mathcal{X}_n) p(\boldsymbol{x})}{\sum_{\boldsymbol{x} \in \mathcal{X}} S(\boldsymbol{x}_i, \boldsymbol{x}; \mathcal{X}_n) p(\boldsymbol{x})}. \tag{24}$$

Eq. (24) use the difference between a FST scenario $\boldsymbol{x}_i$ and the other scenarios in scenario space weighted by the similarity and exposure frequency to estimate $P_{\Delta m}(A|\boldsymbol{x}_i)$. It is practically reasonable if the performance on $\boldsymbol{x}_i$ is significantly different from $\boldsymbol{x}$ with large similarity to it, there may be a potentially large error using the SMs on $\boldsymbol{x}_i$ and $P_{\Delta m}(A|\boldsymbol{x}_i)$ may also be large.

Replace $P_{\Delta m}(A|\boldsymbol{x}_i)$ with $F(\boldsymbol{x}_i; \mathcal{X}_n)$ in Eq. (23) and we will get an approximation of $\tilde{\mu}_{\Delta m,\text{FST}}$, denoted as $\hat{\mu}_{F,\text{FST}}$. Ignoring the constant item and assigning a weight parameter to balance the contributions of the original error and the additional error, the optimization target in Eq. (22) is written as

$$\begin{aligned} &\min_{\mathcal{X}_n} \ L(\mathcal{X}_n) \\ &\text{s.t. } L(\mathcal{X}_n) = \max_{m \in \mathcal{M}} \{|\tilde{\mu}_{\text{FST}} - \mu_m|\} + w_F |\hat{\mu}_{F,\text{FST}}|. \end{aligned} \tag{25}$$

w_F is the weight of the fluctuation estimator. If we have confidence on the SM set $\mathcal{M}$ and have assumption $m^* \in \mathcal{M}$, w_F is set to 0. The optimization is conducted in the same way as described in Section IV-D.

V. Experiment

A. Cut-in Scenario

In this section we use the simulation experiment in cut-in scenario to verify the proposed FST method. Cut-in (as depicted in Fig. 3) is a simple and common scenario for AV testing [13], [14], [15]. The background vehicle (BV) changes lanes ahead of the AV in this scenario, causing risks of collision. The state space of cut-in scenario is simplified as a 2-dimensional variable

$$\boldsymbol{x} = [R, \dot{R}], R \in [0, 90]m, \dot{R} \in [-20, 10]m/s, \tag{26}$$

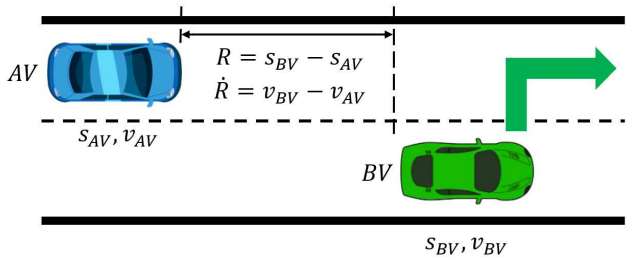

Fig. 3. Cut-in scenario. s and v denote the longitudinal position and velocity. R and $\dot{R}$ denote the range and range rate, respectively.

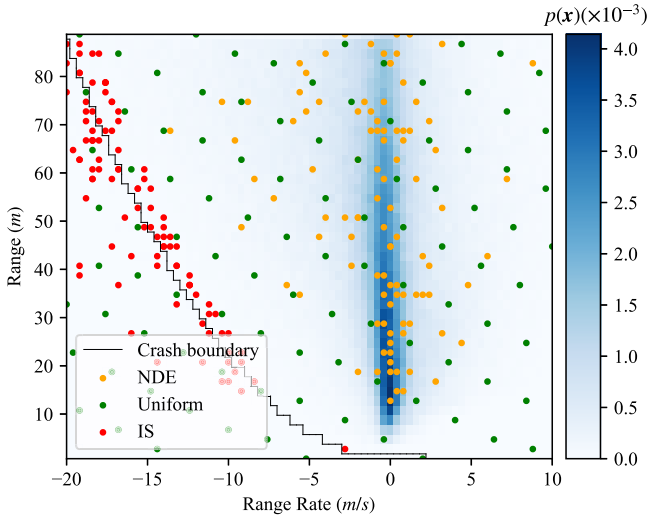

Fig. 4. Illustration of scenarios generated by baseline testing methods in the cut-in state space.

where R and $\dot{R}$ denote the range and range rate at the moment of lane-changing maneuver. By collecting driving behaviors of human drivers in NDE, we can derive the exposure frequency of scenarios as $p(\boldsymbol{x})$. In this experiment we focus on the crash rate and take crash event as A. Then if a collision occurs between AV and BV, we have $P(A|\boldsymbol{x}) = 1$, otherwise $P(A|\boldsymbol{x}) = 0$. We initialize and run the cut-in scenario in simulation to get the performance of AVs.

The BVs in our experiment are 4 intelligent driver models (IDM) (denoted as $m_1, \ldots, m_4$) with different parameters. $\mathcal{M}$ is constructed according to Eq. (8). We use another 3 IDMs and a full velocity difference model (FVDM) [37] as AVs to verify the FST method. The crash rates of these AVs are 2.97×10^{-3}, 1.42×10^{-3}, 6.64×10^{-4}, 1.55×10^{-3}, and the AVs are denoted as AV-1 to AV-4, respectively. The crash boundary of AV-1 and the exposure frequency of scenarios are shown in Fig. 4. Scenarios with smaller R and $\dot{R}$ are of higher risks and may cause crashes.

We use CMC, random quasi-Monte Carlo (RQMC) method [38] and importance sampling (IS) method [13] as baselines. The results of these methods with 100 samples are also shown in Fig. 4. In this paper, we refer to CMC as NDE testing and RQMC as uniform testing, respectively. Scenarios of NDE gather in areas with high exposure frequency and the crash scenarios are hardly concerned, resulting in the low efficiency. IS samples critical scenarios with high risks, but the critical scenarios are similar, which cannot make the most of information with small testing budgets.

B. Qualitative Analysis

First we discuss the qualitative performance of FST method. We set the FST budget $n = 10$ and trained the similarity network with $\mathcal{M}$. Because of the complexity of temporal simulation, the performance of AV $P_{m^*}(A|\boldsymbol{x})$ was hard to be directly represented as a linear combination of BVs and we set the optimization parameter $w_F = 1$.

The basic idea of the FST method is to extract the information of state space provided by SMs leveraging the scenario similarity. We use an example of similarity map to illustrate the similarity learned in our experiment in Fig. 5. The crash boundaries of 4 SMs are shown in the black curve. We use colors to represent the normalized similarity between a FST scenario $\boldsymbol{x}_i$ and the scenario space, i.e. $S(\boldsymbol{x}_i, \boldsymbol{x}; \mathcal{X}_n)$. We inspect different scenarios in the FST scenario set and draw 9 similarity maps of 10 scenarios. FST scenarios within the safe, high-frequency regions of the state space tend to receive large weights $w(\boldsymbol{x}; \mathcal{X}_n)$ because of high similarity values. In contrast, scenarios with higher risks or notable performance differences among SMs are assigned lower weights. Moreover, subtle performance differences near the crash boundaries of SMs are also captured by the similarity.

For further verification of the FST method, We applied the same similarity network on testing tasks with $n = 5, 10, 20$ and searched for the optimal FST scenario set. Fig. 6 shows the examples of the FST scenario set and the similarity of the scenarios. In all three cases with different testing budgets, only a small percentage of testing scenarios are in high-probability safe scenarios, which means sufficient information in this area is gained with a small number of scenarios. These scenarios are needed since unknown unsafe scenarios may occur in this region. As a contrast, most of the testing resources are corner cases allocated to areas near the crash boundaries. In these areas, the performance of the potential AV is supposed to be more uncertain, thus requiring more tests to get accurate results. We assign different colors to scenarios in the state space based on their maximum similarity to the FST scenarios. Remarkably, the dividing lines between FST scenarios are close to the accident boundary of SMs, indicating that the structure of SMs' state space can be learned by our method.

C. Quantitative Testing and Evaluation Results

In order to quantify the efficiency of the FST method, we use NDE testing, uniform testing, importance sampling and the previous coverage-based FST (FST-C) [33] as baselines to carry out tests on AVs. Since the testing and evaluation result is deterministic with the same FST scenarios set, we introduce randomness to our method by randomly initializing the scenarios for optimization. We used these 5 methods to test AV-1 $\sim$ 4 with $n = 5, 10, 20$, which are limited budgets, and obtained the average error and variance. To verify accuracy in terms of maximum error, we sorted the testing and evaluation results to get the maximum error with a confidence level of

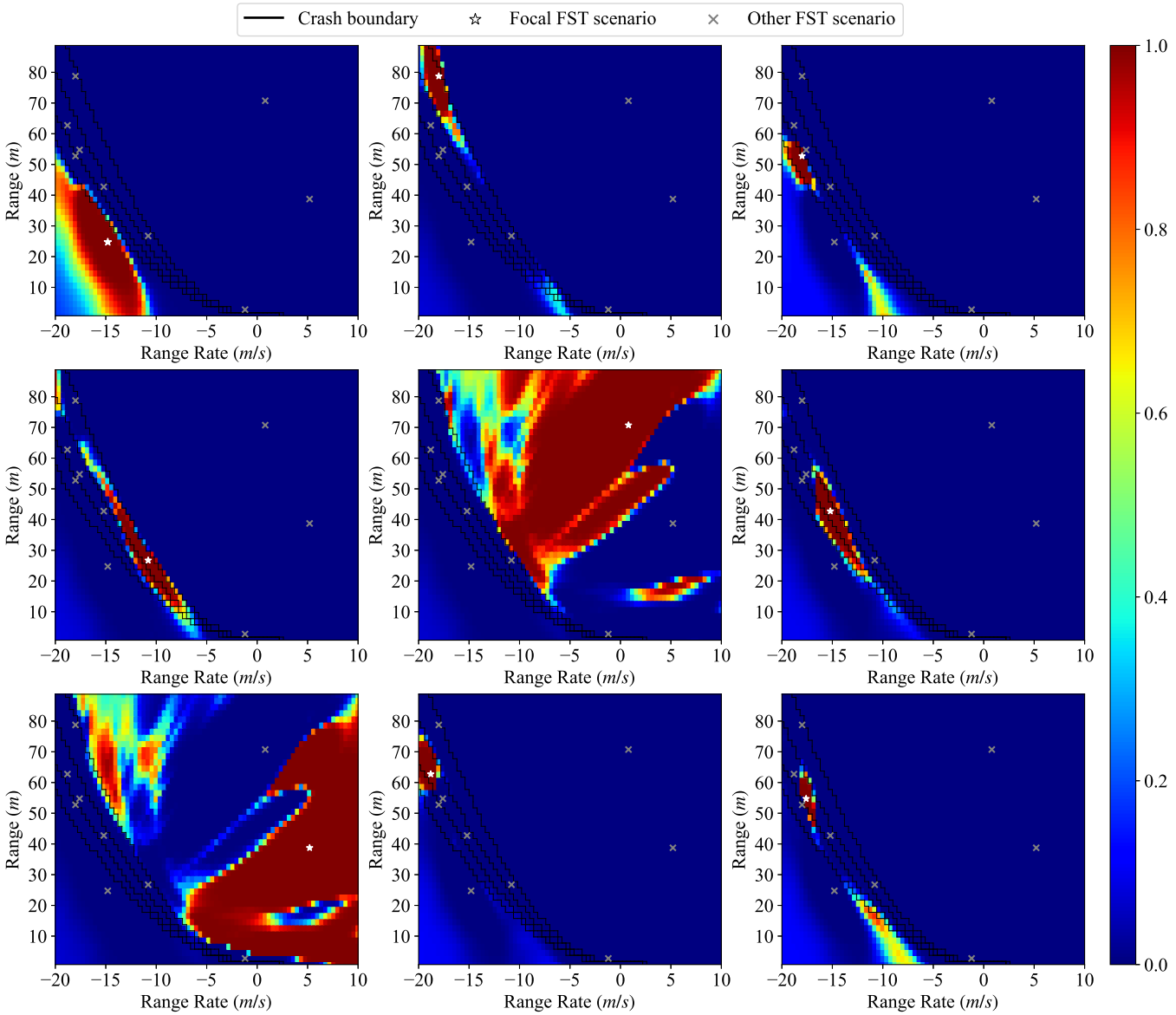

Fig. 5. Examples of FST similarity maps with 10 scenarios. Each scenario has its own similarity map, and 9 of them are shown.

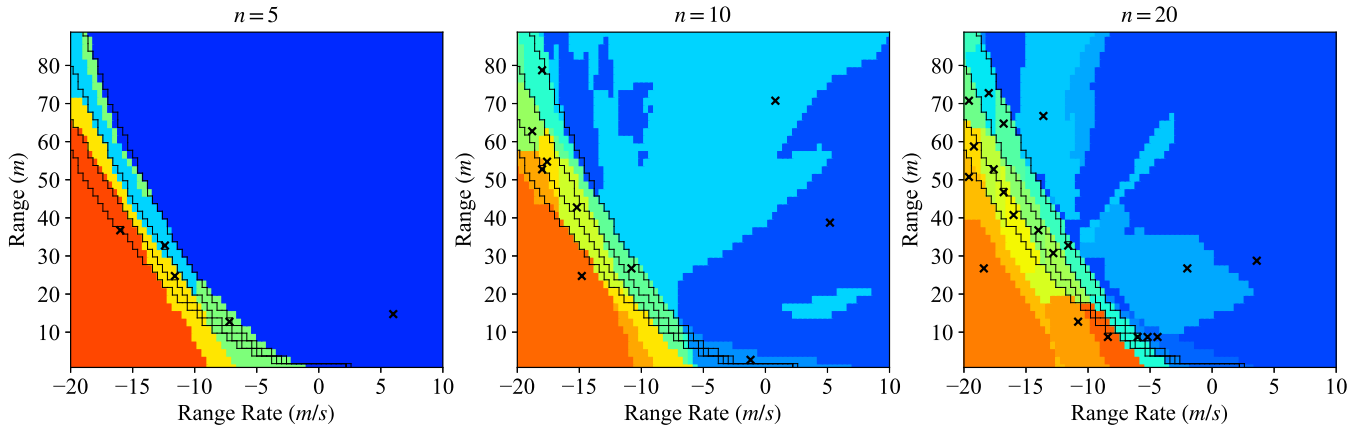

Fig. 6. Example of FST scenarios and the relative similarity. The color of a scenario in state space is determined by the FST scenario with the maximum similarity.

TABLE II

STATISTICS OF TESTING AV-1 ~ 4 WITH $n = 5, 10, 20$ SAMPLES (TESTING ERROR / RELATIVE ERROR)

Method		Average error ($\times 10^{-3}$) ↓			Variance ($\times 10^{-6}$) ↓			Maximum error ($\alpha = 1\%$)($\times 10^{-3}$) ↓		
		$n = 5$	$n = 10$	$n = 20$	$n = 5$	$n = 10$	$n = 20$	$n = 5$	$n = 10$	$n = 20$
AV-1	NDE	5.85/197%	5.77/194%	5.60/188%	592	296	148	197/−*	97.0/−	47.0/−
	Uniform	3.34/113%	2.88/96.9%	1.75/58.7%	21.3	15.5	4.79	18.7/630%	14.5/489%	6.19/208%
	IS	1.12/37.7%	0.74/25.0%	0.53/17.8%	**0.54**	**0.34**	**0.17**	2.97/100%	2.38/80.2%	1.72/58.0%
	FST-C	1.14/38.2%	1.07/36.2%	0.82/27.8%	1.77	1.43	0.94	2.97/100%	2.15/72.4%	1.97/66.4%
	FST	**0.72/24.2%**	**0.61/20.4%**	**0.37/12.6%**	0.59	0.46	0.22	**1.87/63.2%**	**1.06/35.7%**	**0.93/31.3%**
AV-2	NDE	2.82/199%	2.80/197%	2.76/194%	284	142	71.0	1.42/100%	98.6/−	48.6/−
	Uniform	1.77/124%	1.42/99.9%	0.88/61.8%	6.29	3.78	1.49	10.2/718%	6.90/485%	5.66/398%
	IS	0.91/64.1%	0.85/60.4%	0.64/45.5%	0.44	0.40	0.19	3.27/230%	3.27/230%	2.10/148%
	FST-C	0.76/53.7%	0.57/40.3%	0.46/32.6%	1.06	0.66	0.38	2.37/167%	2.34/164%	2.01/141%
	FST	**0.60/42.1%**	**0.38/27.0%**	**0.20/13.9%**	**0.39**	**0.21**	**0.073**	**1.10/77.3%**	**0.83/58.7%**	**0.74/52.0%**
AV-3	NDE	1.32/199%	1.32/199%	1.31/197%	132	66.4	33.2	0.66/100%	0.66/100%	49.3/−
	Uniform	1.08/162%	0.77/116%	0.50/75.7%	2.47	1.06	0.38	6.00/903%	4.33/652%	1.83/276%
	IS	0.73/110.1%	0.46/69.3%	0.34/50.7%	0.17	0.14	0.061	2.85/430%	1.68/253%	1.10/165%
	FST-C	0.66/100%	0.50/74.8%	0.33/50.2%	1.39	0.39	0.20	2.21/333%	1.77/266%	1.35/203%
	FST	**0.23/34.6%**	**0.20/30.8%**	**0.15/23.1%**	**0.071**	**0.049**	**0.029**	**0.63/95.2%**	**0.40/59.9%**	**0.28/42.8%**
AV-4	NDE	3.08/198%	3.05/197%	3.01/194%	310	155	77.4	1.55/100%	98.4/−	48.4/−
	Uniform	1.94/125.1%	1.60/103.0%	0.99/63.7%	3.33	2.14	0.62	11.8/760%	7.60/491%	3.45/222%
	IS	0.94/60.6%	0.65/42.2%	0.45/29.3%	0.42	0.22	0.13	3.14/203%	1.97/127%	1.55/100%
	FST-C	0.80/51.4%	0.53/34.2%	0.43/27.9%	0.41	0.27	0.14	2.12/137%	2.09/135%	1.96/127%
	FST	**0.61/39.6%**	**0.40/26.2%**	**0.22/14.3%**	**0.027**	**0.050**	**0.030**	**1.07/69.1%**	**0.87/55.9%**	**0.69/44.7%**

* The relative error is too large and meaningless thus omitted.

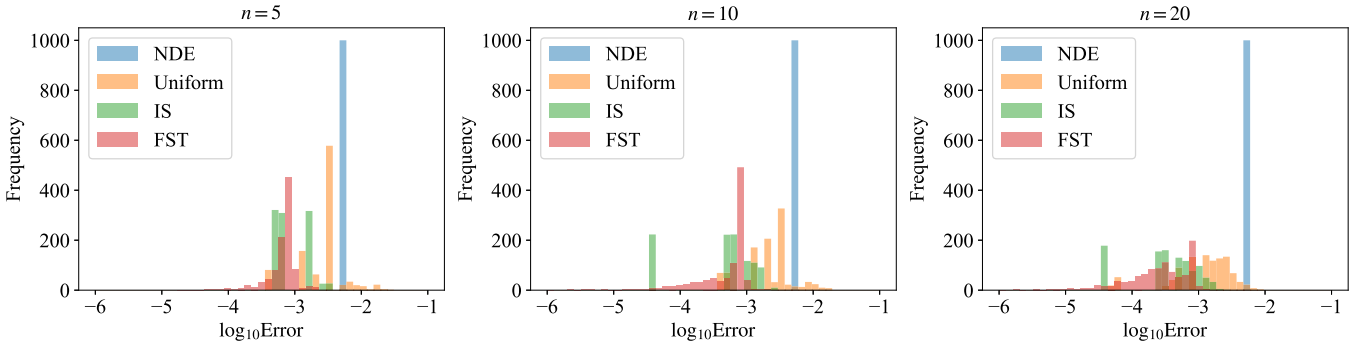

Fig. 7. The comparison of distributions of the estimation errors on AV-1.

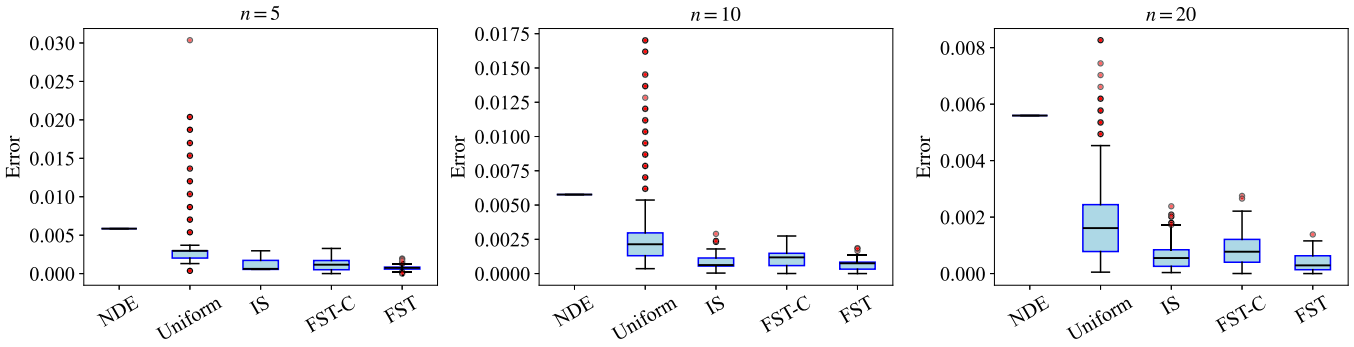

Fig. 8. The comparison of the error box plots on AV-1.

99% (α = 1%). The statistics with the relative error to the ground truth of AVs are listed in TABLE II.

NDE testing faces the problem of "curse of rarity" and the problem is even more serious in our experiment with small n. To generate a valid result for these metrics, a large number of tests must be taken, so we theoretically computed the metrics for NDE testing. The metrics of the other methods are from 1000 repeated tests on AVs. From Table II we can see that the result of NDE is useless with large errors or 100% relative errors (meaning that the evaluation result is 0). Compared to

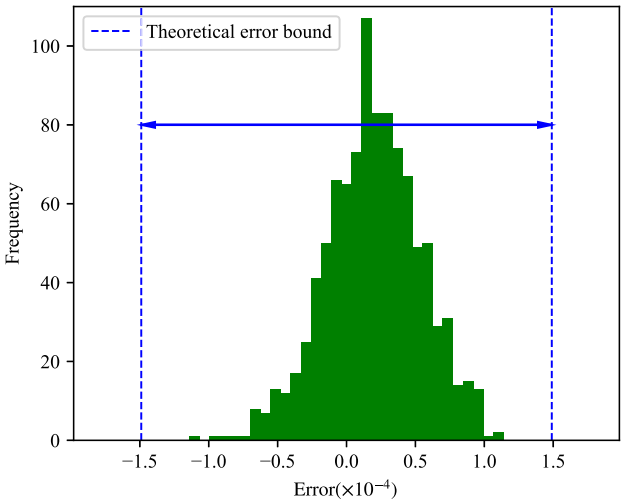

Fig. 9. Illustration of upper bound of evaluation error.

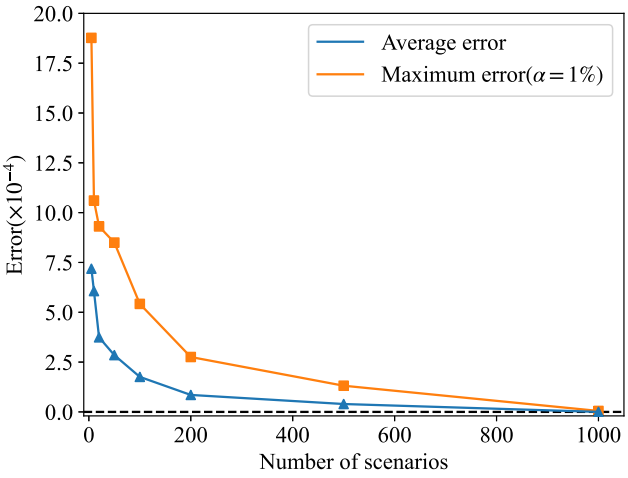

Fig. 10. Results with different number of scenarios.

NDE testing, other baseline methods achieve higher accuracy; however, their maximum error remains unacceptably high. The IS method is highly dependent on the specific AV model and testing budget, demonstrating improved performance with AV-1 or under larger testing budgets. The proposed FST method significantly outperforms other approaches across these metrics. Notably, when the number of tests is small, the accuracy of the FST method (measured by both error and maximum error) is less affected compared to traditional methods. Its relative error remains more acceptable, suggesting realistic availability for rapid and accurate AV testing.

The distributions of logarithmic error of the FST method and baseline methods on AV-1 are also shown in Fig. 7. The box plots of the same experiment setups are shown in Fig. 8. Testing were repeated 1000 times with each method. The stability of optimization can be verified in this experiment. With NDE, most of the crash rates tested out of 1000 results are 0, yielding an evaluation error of 2.97×10^{-3}. Compared to the other baseline methods, the error tested by FST is tightly clustered. It also exhibits a sharp decline in error frequency as the error magnitude increases, highlighting its effectiveness in minimizing the upper bound of error.

D. Ideal Upper Bound of Evaluation Error

In Section IV, we prove that in ideal cases where $m^* \in \mathcal{M}$, a theoretical upper bound of evaluation error is ensured. Here we set $w_F = 0$ and search for a set of 10 optimal FST scenarios. The optimized upper bound of error is 1.49×10^{-4}. We manually sampled AVs from $\mathcal{M}$ to test. The crash rate of AVs varies from 4.62×10^{-4} to 4.90×10^{-3}. These AVs are not real enough but represent the situations where the prior knowledge is relatively accurate. We tested 1000 AVs and the results of the testing and evaluation error are shown in Fig. 9. It can be seen that the evaluation error of all AVs is restricted within the upper bound, and the maximum relative error of the AVs is 32.2%.

E. Ablation Study

We explored the feature of our method in this section. The following experiments were conducted with $n = 10$ and repeated 10^3 times, unless otherwise specified.

Effectiveness of modules in FST: We examined the effect of different modules of FST by testing AV-1 ~ 4 with $n = 10$. We gradually added the optimization module and the additional error reduction module with fluctuation estimator to verify the performance of FST method. The results are listed in Table III. By combining similarity network and optimization strategy, the performance of FST is significantly improved compared with FST-C and randomly initialized FST scenarios. The effects of fluctuation estimator depends on specific AVs. Generally, the maximum evaluation error of FST method can be improved with the additional error reduction scheme.

Testing with larger n: We examined the effect of the number of scenarios n by testing AV-1 with up to 1000 scenarios. Although our method is designed for applications with a strictly limited testing budget, it can achieve higher accuracy with a larger testing budget. The results are shown in Fig. 10. FST method effectively controls the error with as few as 5 scenarios and converges to the ground truth as the number of scenarios increases.

Influence of hyper-parameters: The key parameters of the FST method include k clusters for generating P_c and the weight parameter of fluctuation estimator, w_F. For the number of clusters, we conducted experiments with $k = 1, 2, 5, 10, 20$ and the results are shown in Table IV. When the number of clusters is set to 1, P_c becomes a uniform distribution, which deviates from $\mathcal{X}^*$ and degrades the performance of the similarity network. As k increases, the testing and evaluation error decreases, reaching its minimum at $k = 5$. Regarding the weight parameter w_F, we conducted experiments on the IDM AV-1 and the FVDM AV-4 to assess the impact of the fluctuation estimator across different AVs and the results are shown in Fig. 11 and Fig. 12. Results indicate that the optimal choice of w_F depends on the accuracy of the SM set in representing a specific AV model. For AV-4, the fluctuation estimator reduces the average testing error as AV-4 is dissimilar

TABLE III

ABLATION ON FST WITH $n = 10$ SAMPLES. WHEN OPTIMIZATION MODULE IS DISABLED WE RANDOMLY SAMPLE ALL FST SCENARIOS. w_F IS SET TO 1 IF USING THE FLUCTUATION ESTIMATOR MODULE AND OTHERWISE $w_F = 0$. WE USE THE COVERAGE-BASED METHOD IN [33] (I.E. FST-C) WHEN SIMILARITY NETWORK IS NOT USED

Similarity network	Optimization module	Fluctuation estimator	Average error ($\times 10^{-4}$) ↓				Maximum error ($\alpha = 1\%$)($\times 10^{-4}$) ↓			
			AV-1	AV-2	AV-3	AV-4	AV-1	AV-2	AV-3	AV-4
×	✓	✓	10.7	5.73	4.97	5.30	21.5	23.4	17.7	20.9
✓	×	×	14.9	7.84	4.37	8.17	29.7	19.0	18.3	15.5
✓	✓	×	**5.64**	3.87	2.13	4.24	11.5	8.59	5.27	8.99
✓	✓	✓	6.06	**3.84**	**2.05**	**4.06**	**10.6**	**8.34**	**3.98**	**8.67**

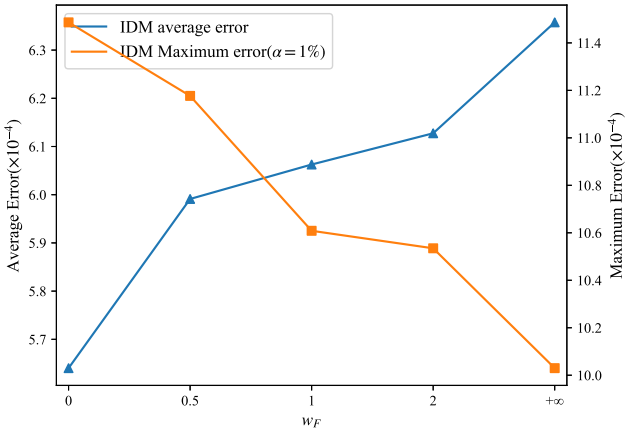

Fig. 11. Testing with different w_F on IDM AV-1.

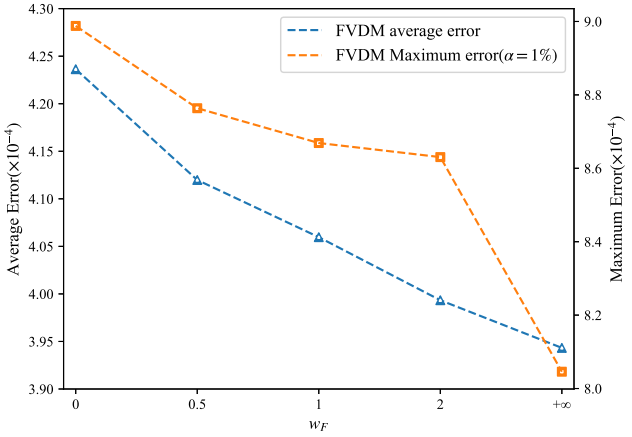

Fig. 12. Testing with different w_F on FVDM AV-4.

TABLE IV

ABLATION ON THE K-MEANS CLUSTERS

cluster number k	Average error ($\times 10^{-4}$) ↓
1	10.6/35.6%
2	7.65/25.8%
5	6.06/20.4%
10	6.09/20.5%
20	6.31/21.2%

to the SMs. In contrast, for AV-1, the impact is reversed. Generally, the experiments demonstrate that the fluctuation estimator effectively reduces the maximum testing error across different AVs. To balance the effect of the fluctuation estimator, we set $w_F = 1$ in our experiments.

TABLE V

ABLATION ON THE STRUCTURE OF SIMILARITY NETWORK

MLP layers	Hidden dimensions	Average error ($\times 10^{-4}$) ↓
4	128	7.02/23.7%
4	256	6.63/22.3%
8	256	6.06/20.4%
12	512	6.09/20.5%

TABLE VI

GENERALIZATION ABILITY OF SIMILARITY NETWORK ON TRAINING AND TESTING DATASET WITH DIFFERENT n

Testing n	Training n	Average error ($\times 10^{-4}$) ↓	Maximum error ($\alpha = 1\%$)($\times 10^{-4}$) ↓
5	5	7.75/26.1%	19.0/64.0%
	10	7.18/24.2%	18.8/63.2%
	20	**7.02/23.6%**	**16.6/55.8%**
10	5	6.67/22.5%	18.9/63.8%
	10	**6.06/20.4%**	10.6/35.7%
	20	6.11/20.6%	**9.49/32.0%**
20	5	3.81/12.8%	9.97/33.6%
	10	3.74/12.6%	9.31/31.3%
	20	**3.31/11.1%**	**8.91/30.0%**

Choices of MLP structure: We examined different structures of MLP as the backbone and performed tests on AV-1. As the results in Table V show, a simple structure will weaken the ability of the network. For the cut-in experiment with a simple MLP structure, the 8-layer MLP is sufficient to effectively extract information from the state space. More advanced network architectures will be explored in future work.

F. Generalization Across Different Budgets

The similarity network is designed to learn the latent feature of scenarios in the state space to determine the weights of FST scenarios. This feature is not strictly related to the number of tests n, as mentioned in Section IV-D. We trained the similarity network using data consisting of $n = 5, 10, 20$ scenarios, respectively, and conducted cross-experiments on AV-1 using $n = 5, 10, 20$. The results are shown in TABLE VI. Generally, after training with a certain number of scenarios, the FST method will be able to generate the FST scenario

set under different budgets. With larger number of training FST scenarios, the similarity network will achieve better performances.

VI. Conclusion

In this paper, we propose the few-shot testing method to tackle the challenge of testing the performance index of AVs with a severely limited testing budget. Existing testing methods suffer from low accuracy and efficiency given a small number of tests, making it practically impossible to quickly obtain accurate testing and evaluation results. We deal with this problem by searching for a fixed few-shot testing scenario set to mitigate the uncertainty resulting from the limited number of tests and minimizing the upper bound of the evaluation error. A similarity network is employed to learn the features of the scenario space using surrogate models. The results show that proposed method significantly improves accuracy in few-shot testing cases and generates for the first time a practically acceptable upper bound of the evaluation error with a certain confidence level. This would bring the possibility for reliable and rapid testing of AVs. We provide simulation experiments in cut-in case in this paper. Theoretically, the FST method can be applied in complex scenarios and real-world AV testing. The application of FST method in real-world interactive scenarios will be an important research direction.

VII. Acknowledgment

The authors acknowledge Wuhan East Lake High-Tech Development Zone (also known as the Optics Valley of China, or OVC) National Comprehensive Experimental Base for Governance of Intelligent Society.

References

[1] L. Li et al., "Artificial intelligence test: A case study of intelligent vehicles," *Artif. Intell. Rev.*, vol. 50, no. 3, pp. 441–465, Oct. 2018.

[2] L. Li et al., "Parallel testing of vehicle intelligence via virtual-real interaction," *Sci. Robot.*, vol. 4, no. 28, p. 4106, Mar. 2019.

[3] X. Li, P. Ye, J. Li, Z. Liu, L. Cao, and F.-Y. Wang, "From features engineering to scenarios engineering for trustworthy AI: I&I, C&C, and V&V," *IEEE Intell. Syst.*, vol. 37, no. 4, pp. 18–26, Jul. 2022.

[4] X. Li, S. Teng, B. Liu, X. Dai, X. Na, and F.-Y. Wang, "Advanced scenario generation for calibration and verification of autonomous vehicles," *IEEE Trans. Intell. Vehicles*, vol. 8, no. 5, pp. 3211–3216, May 2023.

[5] F.-Y. Wang et al., "Verification and validation of intelligent vehicles: Objectives and efforts from China," *IEEE Trans. Intell. Vehicles*, vol. 7, no. 2, pp. 164–169, Jun. 2022.

[6] S. Riedmaier, T. Ponn, D. Ludwig, B. Schick, and F. Diermeyer, "Survey on scenario-based safety assessment of automated vehicles," *IEEE Access*, vol. 8, pp. 87456–87477, 2020.

[7] H. X. Liu and S. Feng, ""Curse of rarity' for autonomous vehicles," 2022, *arXiv:2207.02749*.

[8] S. Feng, Y. Feng, C. Yu, Y. Zhang, and H. X. Liu, "Testing scenario library generation for connected and automated vehicles, part I: Methodology," *IEEE Trans. Intell. Transp. Syst.*, vol. 22, no. 3, pp. 1573–1582, Mar. 2021.

[9] J.-A. Bolte, A. Bar, D. Lipinski, and T. Fingscheidt, "Towards corner case detection for autonomous driving," in *Proc. IEEE Intell. Vehicles Symp. (IV)*, Jun. 2019, pp. 438–445.

[10] G. Bagschik, T. Menzel, and M. Maurer, "Ontology based scene creation for the development of automated vehicles," in *Proc. IEEE Intell. Vehicles Symp. (IV)*, Jun. 2018, pp. 1813–1820.

[11] N. Webb et al., "Waymo's safety methodologies and safety readiness determinations," 2020, *arXiv:2011.00054*.

[12] J. Yang, H. Sun, H. He, Y. Zhang, H. X. Liu, and S. Feng, "Adaptive safety evaluation for connected and automated vehicles with sparse control variates," *IEEE Trans. Intell. Transp. Syst.*, vol. 25, no. 2, pp. 1761–1773, Feb. 2024.

[13] S. Feng, Y. Feng, H. Sun, S. Bao, Y. Zhang, and H. X. Liu, "Testing scenario library generation for connected and automated vehicles, part II: Case studies," *IEEE Trans. Intell. Transp. Syst.*, vol. 22, no. 9, pp. 5635–5647, Sep. 2021.

[14] D. Zhao, X. Huang, H. Peng, H. Lam, and D. J. LeBlanc, "Accelerated evaluation of automated vehicles in car-following maneuvers," *IEEE Trans. Intell. Transp. Syst.*, vol. 19, no. 3, pp. 733–744, Mar. 2018.

[15] S. Zhang, H. Peng, D. Zhao, and H. E. Tseng, "Accelerated evaluation of autonomous vehicles in the lane change scenario based on subset simulation technique," in *Proc. 21st Int. Conf. Intell. Transp. Syst. (ITSC)*, Maui, HI, USA, Nov. 2018, pp. 3935–3940.

[16] X. Zhang et al., "Finding critical scenarios for automated driving systems: A systematic mapping study," *IEEE Trans. Softw. Eng.*, vol. 49, no. 3, pp. 991–1026, Mar. 2023.

[17] F. Kruber, J. Wurst, and M. Botsch, "An unsupervised random forest clustering technique for automatic traffic scenario categorization," in *Proc. 21st Int. Conf. Intell. Transp. Syst. (ITSC)*, Nov. 2018, pp. 2811–2818.

[18] F. Kruber, J. Wurst, E. S. Morales, S. Chakraborty, and M. Botsch, "Unsupervised and supervised learning with the random forest algorithm for traffic scenario clustering and classification," in *Proc. IEEE Intell. Vehicles Symp. (IV)*, Jun. 2019, pp. 2463–2470.

[19] P. Weissensteiner, G. Stettinger, S. Khastgir, and D. Watzenig, "Operational design domain-driven coverage for the safety argumentation of automated vehicles," *IEEE Access*, vol. 11, pp. 12263–12284, 2023.

[20] J. Duan, F. Gao, and Y. He, "Test scenario generation and optimization technology for intelligent driving systems," *IEEE Intell. Transp. Syst. Mag.*, vol. 14, no. 1, pp. 115–127, Jan. 2022.

[21] M. Klischat and M. Althoff, "Generating critical test scenarios for automated vehicles with evolutionary algorithms," in *Proc. IEEE Intell. Vehicles Symp.*, Jun. 2019, pp. 2352–2358.

[22] A. Li, S. Chen, L. Sun, N. Zheng, M. Tomizuka, and W. Zhan, "SceGene: Bio-inspired traffic scenario generation for autonomous driving testing," *IEEE Trans. Intell. Transp. Syst.*, vol. 23, no. 9, pp. 14859–14874, Sep. 2022.

[23] L. Li, W.-L. Huang, Y. Liu, N.-N. Zheng, and F.-Y. Wang, "Intelligence testing for autonomous vehicles: A new approach," *IEEE Trans. Intell. Vehicles*, vol. 1, no. 2, pp. 158–166, Jun. 2016.

[24] J. Ge, J. Zhang, C. Chang, Y. Zhang, D. Yao, and L. Li, "Task-driven controllable scenario generation framework based on AOG," *IEEE Trans. Intell. Transp. Syst.*, vol. 25, no. 6, pp. 6186–6199, Jun. 2024.

[25] E. Pronovost et al., "Scenario diffusion: Controllable driving scenario generation with diffusion," in *Proc. Adv. Neural Inf. Process. Syst.*, 2023, pp. 68873–68894.

[26] H. Sun, S. Feng, X. Yan, and H. X. Liu, "Corner case generation and analysis for safety assessment of autonomous vehicles," *Transp. Res. Record, J. Transp. Res. Board*, vol. 2675, no. 11, pp. 587–600, Nov. 2021.

[27] S. Feng, X. Yan, H. Sun, Y. Feng, and H. X. Liu, "Intelligent driving intelligence test for autonomous vehicles with naturalistic and adversarial environment," *Nature Commun.*, vol. 12, no. 1, p. 748, Feb. 2021.

[28] S. Feng et al., "Dense reinforcement learning for safety validation of autonomous vehicles," *Nature*, vol. 615, no. 7953, pp. 620–627, Mar. 2023.

[29] D. Zhao et al., "Accelerated evaluation of automated vehicles safety in lane-change scenarios based on importance sampling techniques," *IEEE Trans. Intell. Transp. Syst.*, vol. 18, no. 3, pp. 595–607, Mar. 2017.

[30] B. Gangopadhyay, S. Khastgir, S. Dey, P. Dasgupta, G. Montana, and P. Jennings, "Identification of test cases for automated driving systems using Bayesian optimization," in *Proc. IEEE Intell. Transp. Syst. Conf. (ITSC)*, Oct. 2019, pp. 1961–1967.

[31] S. Feng, Y. Feng, H. Sun, Y. Zhang, and H. X. Liu, "Testing scenario library generation for connected and automated vehicles: An adaptive framework," *IEEE Trans. Intell. Transp. Syst.*, vol. 23, no. 2, pp. 1213–1222, Feb. 2022.

[32] J. Yang, R. Bai, H. Ji, Y. Zhang, J. Hu, and S. Feng, "Adaptive testing environment generation for connected and automated vehicles with dense reinforcement learning," 2024, *arXiv:2402.19275*.

[33] S. Li, J. Yang, H. He, Y. Zhang, J. Hu, and S. Feng, "Few-shot scenario testing for autonomous vehicles based on neighborhood coverage and similarity," 2024, *arXiv:2402.01795*.

[34] A. B. Owen. (2013). *Monte Carlo Theory, Methods Examples*. [Online]. Available: https://artowen.su.domains/mc/
[35] L. Fraade-Blanar, M. S. Blumenthal, J. M. Anderson, and N. Kalra, *Measuring Automated Vehicle Safety: Forging a Framework*. Santa Monica, CA, USA: RAND Corporation, 2018.
[36] G. Lou, Y. Deng, X. Zheng, M. Zhang, and T. Zhang, "Testing of autonomous driving systems: Where are we and where should we go?," in *Proc. 30th ACM Joint Eur. Softw. Eng. Conf. Symp. Found. Softw. Eng.*, Nov. 2022, pp. 31–43.
[37] R. Jiang, Q. Wu, and Z. Zhu, "Full velocity difference model for a car-following theory," *Phys. Rev. E, Stat. Phys. Plasmas Fluids Relat. Interdiscip. Top.*, vol. 64, no. 1, Jun. 2001, Art. no. 017101.
[38] A. B. Owen. (2023). *Practical Quasi-Monte Carlo Integration*. [Online]. Available: https://artowen.su.domains/mc/practicalqmc.pdf

Shu Li received the bachelor's degree from the Department of Automation, Tsinghua University, China, in 2023, where he is currently pursuing the Ph.D. degree in control science and engineering. His research interests include testing and evaluation of autonomous vehicles and generation of driving environment.

Honglin He received the B.S. degree from Xiamen University, China, in 2021, and the M.S. degree from Tsinghua University, China, in 2024. His current research interests include scenario generation for autonomous vehicles and embodied AI.

Jingxuan Yang received the bachelor's degree from the School of Mechanical Engineering and Automation, Harbin Institute of Technology, Shenzhen, China, in 2020. He is currently pursuing the Ph.D. degree with the Department of Automation, Tsinghua University, Beijing, China. His current research interests include adaptive testing and evaluation of connected and automated vehicles.

Jianming Hu (Senior Member, IEEE) received the B.E., M.E., and Ph.D. degrees in 1995, 1998, and 2001, respectively. He is currently an Associate Professor with the Department of Automation (DA), Tsinghua University. He has presided and participated in more than 20 research projects granted from the Ministry of Science and Technology of China, National Science Foundation of China, and other large companies with more than 30 journal articles and more than 100 conference papers. His research interests include networked traffic flow, large-scale traffic information processing, intelligent vehicle infrastructure cooperation systems (V2X or connected vehicles), and urban traffic signal control.

Yi Zhang (Senior Member, IEEE) received the B.S. and M.S. degrees from Tsinghua University, China, in 1986 and 1988, respectively, and the Ph.D. degree from the University of Strathclyde, U.K., in 1995. He is currently a Professor of control science and engineering with Tsinghua University. His current research interests include intelligent transportation systems. His active research areas include intelligent vehicle-infrastructure cooperative systems, analysis of urban transportation systems, urban road network management, traffic data fusion and dissemination, and urban traffic control and management. His research fields also cover the advanced control theory and applications, advanced detection and measurement, and systems engineering.

Shuo Feng (Member, IEEE) received the bachelor's and Ph.D. degrees from the Department of Automation, Tsinghua University, China, in 2014 and 2019, respectively. He was a Post-Doctoral Research Fellow with the Department of Civil and Environmental Engineering and also an Assistant Research Scientist with the University of Michigan Transportation Research Institute (UMTRI), University of Michigan, Ann Arbor. He is currently an Associate Professor with the Department of Automation, Tsinghua University. His research interests include the development and validation of safety-critical machine learning, particularly for connected and automated vehicles. He was a recipient of the Best Ph.D. Dissertation Award from the IEEE Intelligent Transportation Systems Society in 2020 and the ITS Best Paper Award from the INFORMS TSL society in 2021. He is an Associate Editor of IEEE TRANSACTIONS ON INTELLIGENT VEHICLES and an Academic Editor of the *Automotive Innovation*.